

**PATENT APPLICATION**

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re application of

Docket No: A8506

William R. BELKNAP, et al.

IBM Docket No: SVL920010059US1

Appln. No.: 09/986,248

Group Art Unit: 2173

Confirmation No.: 5036

Examiner: Dennis G. BONSHOCK

Filed: November 8, 2001

For: **SYSTEM AND METHOD FOR PACKING OF SMALL OBJECTS FOR EFFICIENT  
DELIVERY OVER NETWORKS**

**SUBMISSION OF SUPPLEMENTAL APPEAL BRIEF**  
**UNDER 37 C.F.R. § 41.37**

**MAIL STOP APPEAL BRIEF - PATENTS**

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir:

In response to the Order Returning the Undocketed Appeal to the Examiner dated January 30, 2009, Applicants submit herewith a supplemental brief to expedite prosecution. The supplemental brief includes:

(i) a revised version of the Arguments (VII) section which deletes the listing of canceled claims 6 and 18 in the statement of rejection.

Although Applicant believes that no fee is due, the USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account.

Respectfully submitted,

/Quadeer A. Ahmed/

SUGHRUE MION, PLLC  
Telephone: (202) 293-7060  
Facsimile: (202) 293-7860

---

Quadeer A. Ahmed  
Registration No. 60,835

WASHINGTON OFFICE  
**23373**  
CUSTOMER NUMBER

Date: March 26, 2009

**PATENT APPLICATION**

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re application of

Docket No: A8506

William R. BELKNAP, et al.

IBM Docket No: SVL920010059US1

Appln. No.: 09/986,248

Group Art Unit: 2173

Confirmation No.: 5036

Examiner: Dennis G. BONSHOCK

Filed: November 8, 2001

For: SYSTEM AND METHOD FOR PACKING OF SMALL OBJECTS FOR EFFICIENT  
DELIVERY OVER NETWORKS

**SUPPLEMENTAL APPEAL BRIEF UNDER 37 C.F.R. § 41.37**

**MAIL STOP APPEAL BRIEF - PATENTS**

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 41.37, Appellant submits the following:

**Table of Contents**

I. REAL PARTY IN INTEREST .....	2
II. RELATED APPEALS AND INTERFERENCES.....	3
III. STATUS OF CLAIMS .....	4
IV. STATUS OF AMENDMENTS .....	5
V. SUMMARY OF THE CLAIMED SUBJECT MATTER .....	6
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL .....	16
VII. ARGUMENT.....	17
CLAIMS APPENDIX.....	29
EVIDENCE APPENDIX:.....	38
RELATED PROCEEDINGS APPENDIX .....	39

**I. REAL PARTY IN INTEREST**

The real party in interest is INTERNATIONAL BUSINESS MACHINES CORPORATION by virtue of an assignment executed by William Russell Belknap and Steven Victor Kauffman (hereinafter, "Appellant") on November 6, 2001, and recorded on November 8, 2001 at Reel 012299, Frame 0950.

**II. RELATED APPEALS AND INTERFERENCES**

To the best of the knowledge and belief of the Appellant, the Assignee, and the undersigned, there are no other appeals or interferences before the Board of Appeals and Interferences (“the Board”) that will directly affect, or be affected by, the Board’s decision in the present Appeal.

**III. STATUS OF CLAIMS**

Claims 1-3, 7-10, 13-15, 19-23, 25-29, 31, 32, and 34-36 are all the claims pending in the application. Claims 1-3, 7-10, 13-15, 19-23, 25-29, 31, 32, and 34-36 have been rejected, and are the subject of this appeal. Claims 4-6, 11, 12, 16-18, 24, 30, and 33 have been canceled without prejudice or disclaimer.

**IV. STATUS OF AMENDMENTS**

The Amendment Under 37 C.F.R. § 1.116, filed February 7, 2008, has been entered. (*See* Advisory Action dated April 29, 2008). Accordingly, there are no outstanding, non-entered amendments of the claims.

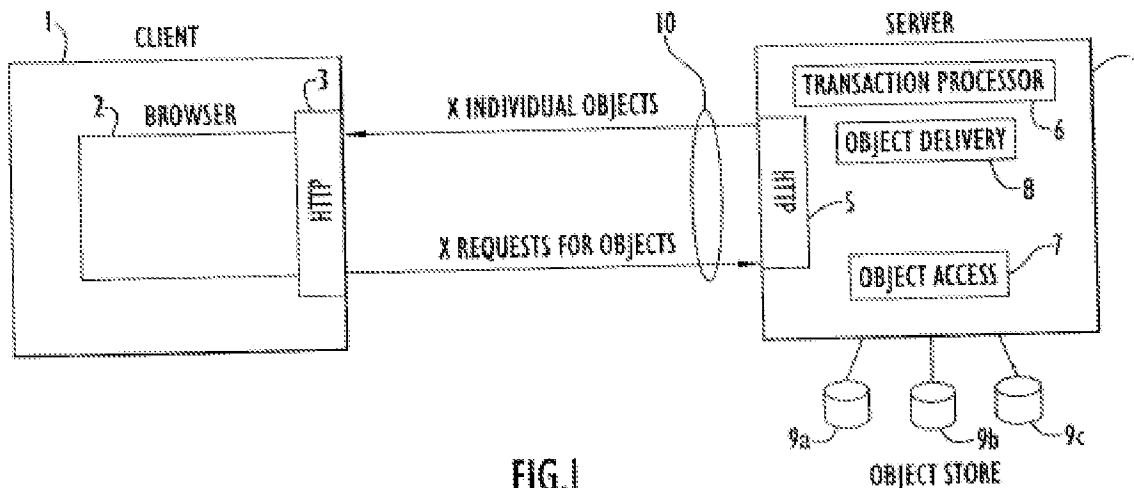
**V. SUMMARY OF THE CLAIMED SUBJECT MATTER**

For the Board's convenience, Appellant will first describe the relevant art (pp. 1-4 of the Specification), and then exemplary embodiments of the invention (pp. 7-16 of the Specification). This discussion of the exemplary embodiments and the pending claims is provided for explanatory purposes only, and is not intended to limit the scope of the claims.

**Relevant Art**

The Internet, which includes the World Wide Web (hereinafter, "web"), has become an important and useful tool for accessing a wide variety of information. A typical website on the web is hosted on a network server computer that includes application software programs. In a conventional client server system (*see* Fig. 1, reproduced below), a client 1 includes a browser 2 and a hypertext transfer protocol (HTTP) module 3 for communicating with a server 4. The server 4 includes an HTTP module 5 for handling the communication sessions between the client and server, and is similar to the client HTTP module 3. The client sends requests for objects by way of the HTTP modules 3 and 5. A transaction processor 6 within the server receives the transaction request and processes it to determine the subject matter of the request. The transaction processor 6 interacts with object access module 7. The object access module receives a request from the transaction processor for a particular object stored on one or more of a plurality of object stores 9a-9c. The object access module 7 retrieves the requested object and provides it to an object delivery module 8 within the server. The object delivery module 8 formats the object and prepares it for delivery over a communication path 10, using, *e.g.*, a TCP/IP session, by way of the HTTP modules 5 and 3. The individual object requested by the client is sent from the server to the client by way of the HTTP module 3 and provided to browser 2 for display to a user. A user, through the client's browser, can request a web page stored at the

server in one of the object stores. The server then locates and returns the web page to the client for display on the browser.



In conventional systems, objects are displayed on the browser in the order they are received from the server because they are not delivered to the browser in a single bundle. (Specification at p. 11, lines 14-15). Such an order, however, is not necessarily the order in which the objects are intended to be displayed. (Specification at p. 11, lines 15-16). Accordingly, there has been a long felt need to control the order in which objects are displayed on a browser.

Further, in conventional systems, objects packed at the server are not automatically unpacked at the client (Specification at p. 14, lines 18-21), thus requiring user interaction to unpack objects sent from a server. Moreover, in the conventional systems, multiple communication sessions must be torn down, resulting in a high degree of overhead processing (Specification, at p. 3, line 16 to p. 4, line 18).

#### Exemplary Embodiments

**Independent claim 1**

Claim 1 is directed to a method of requesting and processing a plurality of objects from a server. Exemplary embodiments of the method are shown in Figs. 3 and 4 and described generally at pages 10-15. The method comprises:

opening a session with at least one server (4); (page 11, lines 20-21)

searching in the at least one server for an information element based upon a search criteria after the opening of the session with the at least one server; (page 11, lines 20-21)

receiving from at least one server search results displayable on a web page comprising a list identifying occurrences of the information element, wherein at least some of said occurrences of the information element identify objects; (page 11, lines 21-22)

generating for at least two identified objects, requests to the at least one server for obtaining the at least two objects; (page 11, line 22 to page 12, line 2)

packing the plurality of requests for the at least two objects into a packed request message and transmitting the packed request message to the at least one server; (page 12, lines 2-7)

receiving a response message from the at least one server, the response message containing the at least two objects packed into the response message; (page 12, lines 9-14)

ending the session with the at least one server after the receiving the response message; (page 12, lines 16-17) and

automatically unpacking the at least two objects contained in the response message for displaying on the web page (page 12, lines 17-18, and page 14, lines 13-21).

**Independent claim 8**

Claim 8 is directed to a method of transferring a plurality of objects from a server to a client. Embodiments of the method are shown in Figs. 3 and 4 and described generally at pages 10-15. The method comprises:

receiving a request from the client (1) for the plurality of objects, wherein the plurality of objects are occurrences of an information element provided as a search criteria in a data network and wherein the request, displayable on a web page, comprises a plurality of requests for the respective plurality of objects; (page 11, line 22 to page 12, line 2)

retrieving the plurality of requested objects from one or more object stores (9a-9c); (page 10, lines 13-14)

automatically packing the retrieved plurality of objects into a response message (page 12, lines 10-13); and

transmitting the response message to the client, wherein the response message comprises an indicator of a first order in which the packed objects are to be displayed (page 12, lines 13-14, and page 13, lines 16-18),

wherein the first order is different from the order in which the plurality of the objects are transmitted to the client (page 13, line 23 to page 14, line 3).

**Independent claim 13**

Claim 13 is directed to a client processor (1). Exemplary embodiments of the client processor are shown in Figs. 3 and 4 and described generally at pages 10-15. The client processor comprises:

a communications module (20) configured to open a session with at least one server (page 11, lines 20-21), configured to receive a response message from the at least one server, the response message containing a plurality of packed objects, wherein the plurality of packed

objects are occurrences of an information element provided as a search criteria in a data network (page 11, lines 21-22), configured to send a request message to the at least one server after opening the session with the at least one server, wherein the request, displayable on a web page, comprises a plurality of requests for the plurality of objects (page 11, line 22 to page 12, line 2), and configured to end the session after receiving response message; (page 12, lines 16-17) an unpacking module (21) configured to automatically unpack from the response message the plurality of packed objects (page 12, lines 17-18, and page 14, lines 13-21); and a browser (2) coupled to the unpacking module, configured to display on the web page the plurality of unpacked objects to a user as search results (page 11, lines 8-9, and page 13, lines 19-20).

**Independent claim 20**

Claim 20 is directed to a server processor (4). Exemplary embodiments of the server processor are shown in Figs. 3 and 4 and described generally at pages 10-15. The server processor comprises:

a communication module (5) configured to receive a request message from a client processor (1) for delivery of a plurality of objects, wherein the plurality of objects are occurrences of an information element provided as a search criteria in a data network and wherein the request message, displayable on a web page, comprises plurality of requests for the respective plurality of objects; (page 11, line 22 to page 12, line 2)

a request processor (5) configured to unpack the requests for the plurality of objects from the request message; (page 12, lines 7-8)

an object access module (8) configured to retrieve the plurality of objects requested by the request processor; (page 10, lines 13-14)

an object packing module (31) coupled to the object access module and configured to automatically pack the plurality of objects retrieved by the object access module into a response message; (page 12, lines 10-13) and

an object delivery module (6)coupled to the object packing module and the communication module and configured to output the response message containing the plurality of packed objects to the client processor, (page 12, lines 13-14)

wherein the response message comprises an indicator indicating a first order in which the packed objects are to be presented, the first order different from the order in which the object packing module packs the plurality of the objects (page 13, lines 16-18, and page 13, line 23 to page 14, line 3).

### **Independent claim 25**

Claim 25 is directed to a computer-readable medium of instructions for requesting and processing a plurality of objects from a server. Exemplary embodiments of the computer-readable medium of instructions for requesting and processing a plurality of objects from a server are shown in Figs. 3 and 4 and described generally at pages 10-15. The computer-readable medium of instructions for requesting and processing a plurality of objects from a server comprises:

program operations for opening a session with at least one server; (page 11, lines 20-21)

program instructions for searching in the at least one server for an information element based upon a search criteria after the opening the session with the at least one server; (page 11, lines 20-21)

program instructions for receiving from the at least one web server search results, displayable on a web page, comprising a list identifying occurrences of the information element,

wherein at least some of said occurrences of the information element identify objects; (page 11, lines 21-22)

program instructions for generating for at least two identified objects, requests to the at least one web server for obtaining respective object; (page 11, line 22 to page 12, line 2)

program instructions for packing the plurality of generated requests for the at least two objects into a single request message and transmitting the packed single request message to the at least one web server; (page 12, lines 2-7)

program instructions for receiving a single response message from the server, the response message containing the at least two objects packed into the response message; (page 12, lines 9-14)

program instructions for ending the session with the at least one server after the receiving the single response message; (page 12, lines 16-17) and

program instructions for automatically unpacking the at least two objects contained in the response message (page 12, lines 17-18, and page 14, lines 13-21).

### **Independent claim 28**

Claim 28 is directed to a computer-readable medium of instructions for transferring a plurality of objects from a server (4) to a client (1). Exemplary embodiments of the computer-readable medium of instructions for transferring a plurality of objects from a server to a client are shown in Figs. 3 and 4 and described generally at pages 10-15. The computer-readable medium of instructions for transferring a plurality of objects from a server to a client comprises:

program instructions for receiving a request from the client for the plurality of objects, wherein the plurality of objects are displayable on a web page as search results and are occurrences of an information element provided as a search criteria in a data network and

wherein the request message comprises a plurality of requests for the respective plurality of objects; (page 11, line 22 to page 12, line 2)

program instructions for retrieving the plurality of requested objects from one or more object stores; (page 10, lines 13-14)

program instructions for automatically packing the retrieved plurality of objects into a response message; (page 12, lines 10-13) and

program instructions for transmitting the response message to the client (page 12, lines 13-14),

wherein the response message comprises an indicator indicating a first order in which the packed objects are to be presented, (page 13, lines 16-18),

wherein the first order is different from the order in which the plurality of the objects are received by the client (page 13, line 23 to page 14, line 3).

### **Independent claim 31**

Claim 31 is directed to a method of transferring a plurality of objects from a server to a client. Embodiments of the method are shown in Figs. 3 and 4 and described generally at pages 10-15. The method comprises:

receiving a request from the client for the plurality of objects, wherein the plurality of objects are displayable on a web page and are occurrences of an information element provided as a search criteria in a data network and wherein the request message comprises a plurality of requests for the respective plurality of objects; (page 11, line 22 to page 12, line 2)

retrieving from an object store a packed object having a plurality of objects corresponding to the requested plurality of objects, wherein the plurality of objects are packed into the packed object prior to receiving the request for the plurality of objects and wherein the

response message comprises an indicator of a predetermined order in which the packed objects are to be presented; (page 15, line 3-17, and page 13, lines 16-18) and transmitting the packed object in a response message to the client (page 12, lines 13-14).

**Independent claim 36**

Claim 36 is directed to a method of requesting and processing a plurality of objects from a server. Exemplary embodiments of the method are shown in Figs. 3 and 4 and described generally at pages 10-15. The method comprises:

opening a session with at least one server; (page 11, lines 20-21)  
receiving from the at least one web server request results comprising a list identifying occurrences of an information element after the opening of the session with the at least one server, wherein at least some of said occurrences of the information element identify image objects and wherein the request results are displayable on a web page; (page 9, lines 15-17, and page 11, lines 21-22)

generating for at least two identified image objects, requests to the at least one web server for obtaining the respective image object; (page 11, line 22 to page 12, line 2)

packing the plurality of requests for the at least two objects into a packed request message and transmitting the packed request message to the at least one web server; (page 12, lines 2-7)

receiving a response message from the at least one web server, the response message containing the at least two objects packed into the response message; (page 12, lines 9-14)

ending the session with the at least one server after the receiving the response message; (page 12, lines 16-17) and

automatically unpacking the plurality of objects contained in the response message (page 12, lines 17-18, and page 14, lines 13-21).

**Dependent claim 7**

Dependent claim 7 depends from claim 1, and recites that the method of claim 1 further comprises displaying on a browser the at least two unpacked objects in a second order provided in the response message, wherein the second order is different from a first order of receipt of the objects (page 13, line 16 to page 14, line 3).

**Dependent claim 19**

Dependent claim 19 depends from claim 13, and recites that the method further comprises displaying on the browser in the web page to the user as the search results, the plurality of unpacked objects in a second order provided in the response message, wherein the second order is different from a first order in which the plurality of the packed objects are unpacked from the response message (page 13, line 16 to page 14, line 3).

**Dependent claim 27**

Dependent claim 27 depends from claim 25, and recites that the program instructions further comprise program instructions for outputting the at least two unpacked objects in a second order indicated in the response message, wherein the second order is different from a first order of receipt of the objects (page 13, line 16 to page 14, line 3).

**VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL**

The following is a concise statement of each ground of rejection presented for review:

1. Whether claims 1-3, 7-10, 13-15, 19-23, 25-29, and 34-36 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,282,711 to Halpern *et al.* (“Halpern”) in view of the Appellants’ admitted prior art (“AAPA”) and U.S. Patent No. 7,099,950 to Jones *et al.* (“Jones”).
2. Whether claims 31 and 32 are rejected properly under 35 U.S.C. § 103(a) as being unpatentable over Halpern in view of U.S. Patent No. 6,075,943 to Feinman, and further in view of the AAPA.

## **VII. ARGUMENT**

### ***Claim Rejections - 35 U.S.C. § 103***

#### **Claims 1-3, 7-10, 13-15, 19-23, 25-29, and 34-36**

Claims 1-3, 7-10, 13-15, 19-23, 25-29, and 34-36 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over U.S. Patent No. 6,282,711 to Halpern *et al.* (“Halpern”) in view of the Appellants’ admitted prior art (“AAPA”) and U.S. Patent No. 7,099,950 to Jones *et al.* (“Jones”). For *at least* the following reasons, Appellants respectfully traverse the rejection.

Appellants respectfully submit that claim 1 is patentable over the proposed combination of Halpern, the AAPA, and Jones. For example, amended claim 1 relates to a method of requesting and processing a plurality of objects from a server. The method comprises, *inter alia*:

- **opening a session with at least one server,**
- *searching* in the at least one server for an information element based upon a search criteria *after the opening of the session with the at least one server,*
- receiving from at least one server search results displayable on a web page comprising a list identifying occurrences of the information element, wherein at least some of said occurrences of the information element identify objects,
- *generating* for at least two identified objects, requests to the at least one server for obtaining the at least two objects,
- *packing* the plurality of requests for the at least two objects into a packed request message and transmitting the packed request message to the at least one server,
- *receiving* a response message from the at least one server, the response message containing the at least two objects packed into the response message,
- **ending the session with the at least one server after the receiving the response message,**
- **automatically unpacking the at least two objects contained in the response message for displaying on the webpage.**

In the Amendment filed on October 16, 2007 (hereinafter, the “October 16<sup>th</sup> Amendment”), Appellants submitted that the combined teachings of Halpern and the AAPA do not teach or suggest that the searching, receiving search results, generating, packing, and receiving the response message are all carried out within a session opened with the at least one server (previous Amendment, pages 17-18).

In the Final Office Action dated December 7, 2007, the Examiner acknowledges that Halpern and the AAPA do not disclose this feature. However, the Examiner asserts that Jones discloses a system in which a client and a server exchange requests and responses, respectively, within a session (Final Office Action, page 5, last paragraph). The Examiner further contends that it would have been obvious for a skilled artisan to modify the server query system of Halpern and the AAPA to include the session initiation and termination taught by Jones because “this allows the client to set up distinct times to use the servers services and maintain the networked connection” (Office Action, page 6, lines 3-8). Appellants respectfully disagree and submit that the Examiner is impermissibly using hindsight in view of the Appellants’ own disclosure in an effort to render the claimed method unpatentable.

For instance, the initial burden of establishing that a claimed invention is *prima facie* obvious rests on the USPTO. *In re Rikckaert*, 9 F.3d 1531, 1532 (Fed. Cir. 1993). To make its *prima facie* case of obviousness, the USPTO must satisfy three requirements (*see* MPEP § 2142; *see also* Examination Guidelines for Determining Obviousness Under 35 U.S.C. 103 in view of the Supreme Court Decision in *KSR International Co. v. Teleflex Inc.*, 72 FR 57526):

- (1) The prior art relied upon, or the knowledge generally available in the art at the time of the invention, must support some apparent reason as to why an artisan of ordinary skill would have found it obvious to modify a reference or to combine references.

This reason should be made explicit. *KSR Int'l Co. v. Teleflex, Inc.*, 127 S. Ct. 1727 (U.S. 2007). *See also* Examination Guidelines at 57528.

- (2) The proposed modification of the prior art must have had a reasonable expectation of success, and that is determined from the vantage point of the artisan at the time the invention was made. Examination Guidelines at 57534; *Amgen, Inc. v. Chugai Pharm. Co.*, 927 F.2d 1200, 1209 (Fed. Cir. 1991).

(3) The prior art reference or combination of references must teach or suggest all the limitations of the claims, or reasons must be given why the difference(s) between the prior art and the claimed invention would have been obvious to one of ordinary skill in the art. *See* Examination Guidelines at 57528; *KSR Int'l Co.* at 1741; *In re Vaeck*, 20 U.S.P.Q.2d 1438, 1442 (Fed. Cir. 1991); *In re Wilson*, 424 F.2d 1382, 1385 (CCPA 1970).

Moreover, whether a motivation to combine prior art references has been demonstrated is a question of fact. *Winner Int'l Royalty Corp. v. Wang*, 202 F.3d 1340, 1348 53 USPQ2d 1580, 1586 (Fed. Cir. 2000). To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references." *Ex parte Clapp*, 227 USPQ 972, 973 (Bd. Pat. App. & Inter. 1985). Otherwise, the conclusion to be reached is that the motivation is predicated on hindsight.

In particular, there must be some showing of the obviousness of the claim as a whole, not the discrete parts to establish *prima facie* obviousness. The tests of whether to combine references needs to be applied rigorously. *See: In re Dembiczaik*, 175 F.3d 994, 999, 50 USPQ2d 1614, 1617 (Fed. Cir. 1999), *limited on other grounds by In re Gartside*, 203 F.3d 1305, 53 USPQ2d 1769 (2000). The same principle applies here because the Examiner's analysis is backward to achieve the end point already defined – carrying out the claimed steps of searching, receiving search results, generating, packing, and receiving the response message all within a single session. The first, is that *prima facie* obviousness is a legal requirement and the burden is on the Examiner to demonstrate using only objective evidence or suggestion from the applied prior art and knowledge in the art, that one of ordinary skill would have been led to the claimed

invention as a whole without recourse to Appellants' disclosure as noted above. See also: *In re Oetiker*, 977 F.2d1443, 1447-48, 24USPQ2d 1443, 1446-47(Fed.Cir.1992); *In re Fine* 837 F.2d1071, 1074-75, 5USPQ2d 1596, 1598-1600(Fed.Cir.1988).

As a matter of law then, it is the burden of the Examiner to demonstrate that the prior art, and not Appellants' disclosure, would lead the skilled artisan to the claimed invention as a whole. What the Examiner has done, and as plainly apparent in the statement of rejection, is to dissect the claim into discrete components and then to apply individual pieces of prior art (See Office Action, pages 5-6). That is the hallmark of hindsight and not the characteristic of obviousness.

For example, the given reason for motivation to combine the teachings of Jones with Halpern and the AAPA is, at best, inadequate. As noted above, the Examiner contends that a person of ordinary skill in the art would modify the server query system of Halpern and the AAPA to include the session initiation and termination taught by Jones because it would allow the client to set up distinct times to use the servers services and maintain the networked connection. However, this motivation of performing the steps set forth in claim 1 within a session is only found in the Appellants disclosure.

Jones is directed to solving the problem of ad hoc syntaxes for specifying requests and replies prevalent in legacy debugging tools related to ASCII protocols (Jones, col. 1, lines 16-47). Jones recognizes that XML can be used to represent both the protocol layer and the application data in a session-oriented protocol. However, it does even remotely suggest or relate to the steps performed in claim 1 between the opening and ending of the session, e.g., searching for an information element, generating requests for at least two objects which were included in the search results for the information element, and thereafter automatically unpacking the at least two objects contained in a response message from the server for displaying the objects on the

web page. Moreover, as the Examiner already admits, Halpern and the AAPA do not disclose or suggest carrying out the above-noted features within a session.

As such, the Examiner's analysis is nothing more than a classic hindsight reconstruction where the claimed invention is debased and trivialized because in retrospect, the Examiner can find the disparate steps, uniquely combined, existing in individual basis in multiple prior art references. Such a hindsight construction has been universally condemned. The Federal Circuit has been unwavering in its condemnation of hindsight logic. Typical, is the decision in *Grain Processing Corp. v. American Maize-Products Co.*, 840 F.2d 902, 907(Fed.Cir.1988) wherein Judge Mayer for the Federal Circuit stated,

Care must be taken to avoid hindsight reconstruction using the patent in suit as a guide through the maze of prior art references combining the right references in the right way so as to achieve the result of the claims in suit.

More recently, Judge Clevenger speaking for the Court on the same issue in *McGinley v. Franklin Sports, Inc.*, 60 USPQ2d 1001, 1008 (Fed. Cir. 2001) stated,

The genius of invention is often a combination of known elements which in hindsight seems preordained. To prevent hindsight invalidation of patent claims, the law requires some "teaching, suggestion, or reason" to combine cited references.

Same here, there is no reason why Halpern, the AAPA, and Jones would stand out from the maze of prior art. Only the Appellants' disclosure (see page 3, line 16 to page 4, line 18) identifies the high degree of overhead processing resulting from the setting up and tearing down of multiple sessions, and the advantages of performing the above-noted claim features within a single session.

In view of the foregoing, Appellants respectfully submit that the Examiner's reasoning to combine these references is based on impermissible hindsight. The Examiner's approach fails to heed the injunction for "guarding against falling victim to the insidious effect of a hindsight syndrome wherein that which only the inventor taught is used against its teacher". *Id.* At 1008.

Here, the Examiner's methodology is nothing more than what has been rejected as the clear application of hindsight. Therefore, Appellants respectfully request withdrawal of the rejection of claim 1 because *at least* the first requirement to establish a *prima facie* case of obviousness has not been met by the Examiner.

Furthermore, even if one skilled in the art were to draw from the teachings of Jones, Appellants respectfully submit that this would still not render claim 1 unpatentable. In claim 1, the session ends after the response message is received. Thereafter, the at least two objects contained in the response message are automatically unpacked for displaying on the webpage. In Jones, there is no explicit disclosure or implicit suggestion as to what operations are carried out after the session is terminated between the server and the client. In fact, even based on a cursory review of FIGS. 2-4 of Jones, it appears that all the operations referencing and acting on the objects are carried out between the initiation and subsequent termination of the session (e.g., any operation on the objects, such as deletion thereof, is carried out within the session). As such, Jones teachings would actually lead a skilled artisan to unpack and display the objects within the session, contrary to the requirements of claim 1. The Examiner has not shown how the combined references teach ending the session after receiving the response, but before automatically unpacking and displaying the objects on a webpage. As such, the third requirement to establish a *prima facie* case of obviousness noted above has also not been satisfied. For *at least* these additional reasons, claim 1 is not unpatentable over the alleged combination of Halpern, the AAPA, and Jones.

In the Advisory Action dated April 29, 2008, the Examiner contends that "both systems request content from the server and receiving data responsive to the request (references are in analogous art), Jones just further teaches requests being [sic] to initiate and terminate a session, along with data requests. One would be motivated to combine this teaching with that of Halpern

and AAPA as distinct times where a session is established helps to maintain network security (see column 2, lines 64 through column 3, line 21)”. However, as pointed out above, this reason of performing the steps set forth in claim 1 within a session is only found in the Appellants disclosure. For example, page 3, line 16 to page 4, line 18 of the Specification identify the high degree of overhead processing resulting from the setting up and tearing down of multiple sessions, and the advantages of performing the above-noted claim features within a single session. Accordingly, the Examiner has failed to establish a *prima facie* case of obviousness.

Since claims 2-3, 7-10, 13-15, 19-23, 25-29, and 34-36 are also rejected based on the combined teachings of Halpern, the AAPA, and Jones, they are patentable for *at least* reasons similar to those given above with respect to claim 1.

Furthermore, with respect to independent claims 8, 20, and 28 the Examiner contends in the Final Office Action that Jones, in col. 11, lines 40-52, discloses that the response message comprises an indicator of a first order in which the packed objects are to be displayed, wherein the first order is different from the order in which the plurality of the objects are automatically packed. Appellants respectfully disagree and submit that the teachings of Jones are being misinterpreted in the Final Office Action.

The cited portion of Jones discloses asynchronous operations in its XML-based protocol. Here, instead of each response from the server being paired directly with the request initiated by the client (“with ordering preserved”), responses are returned in an order different from the order of the requests initiated by the client (Jones, col. 11, lines 37-44). As an initial matter, Appellants point out that Jones does not even disclose that any of these responses includes a plurality of the objects that are automatically **packed**. Furthermore, the order referred to in Jones is the order of discrete responses from the server which are transmitted in response to discrete requests from the client. Jones discloses that this order, i.e., the order in which each of

these individual responses is separately transmitted to the client, may be different from the order of the requests initiated by the client. Jones does not even disclose any information regarding the order in which objects within a response message are packed, let alone disclose that the order in which the objects are displayed is different from the order in which the objects are packed. That is, Jones' ordering relates to the ordering of multiple response messages corresponding to multiple requests, and not to the order of objects within a response message corresponding to a request. On the other hand, the first and the second order recited in the claims relates to the order of objects within a single response message.

In response, in the Advisory Action the Examiner alleges that Jones teaches “responses by the server are returned in an order different from the order of the requests initiated by the client” (Jones, col. 11, lines 40-43), and the AAPA is relied upon for the teaching of “packing”. As an initial matter, Appellants point out that the cited portion in Jones discloses that the subject responses correspond to requests, i.e., multiple requests are transmitted to the server in Jones. On the other hand, the claimed plurality of requests are packed into a request message, and the single packed request message is transmitted to the server. As such, this portion of Jones cannot disclose or suggest that the first order included in a response message is different from the order in which the plurality of the objects are automatically packed as set forth in claim 8. Moreover, the Examiner’s referral to the AAPA for teaching the packing is not plausible in this case since this would require Jones’ server to collect requests for a given time. Jones however, does not even remotely suggest such a server configuration. Instead, as discussion above, it only discloses that the server is capable of asynchronous operation (in terms of requests and responses) to allow multiple commands to be processed in parallel. There is no hold-up of requests in Jones which would be necessary to implement the AAPA’s alleged packing procedure.

Therefore, Appellants respectfully submit that claims 8, 20, and 28 are patentable over any conceivable combination of Halpern, the AAPA, and Jones. Accordingly, Appellants respectfully request the Examiner to withdraw the 35 U.S.C. § 103(a) rejection.

**Claims 31 and 32**

Claims 31 and 32 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Halpern in view of U.S. Patent No. 6,075,943 to Feinman, and further in view of the AAPA. For *at least* the following reasons, Appellants respectfully traverse the rejection.

Claim 31 relates to a method of transferring a plurality of objects from a server to a client. The method comprises, *inter alia*, *receiving* a request from the client for the plurality of objects, and *retrieving* from an object store a packed object having a plurality of objects corresponding to the requested plurality of objects. The plurality of objects requested for by the client are packed into the packed object prior to receiving the request for the plurality of objects. The response message comprises an indicator of a predetermined order in which the packed objects are to be presented. In the Final Office Action, the Examiner contends that cols. 2, 3, and 5 of Feinman disclose the above-noted features of claim 31. Appellants respectfully disagree.

For instance, the Examiner, citing col. 5, lines 49-55 of Feinman, contends that since the automatic installations system builds a command for the remote submission, the command (indication) containing the name of the appropriate decompression program to run specifies the order to present the data (Office Action, page 27). However, as submitted on pages 19-21 of the October 16<sup>th</sup> Amendment, Appellants point out that even if the installation of application programs suggests displaying the application programs, the compressed file sent to the client in Feinman does not comprise the sequential file 100 which is used to identify a remote client's delivery points and delivery timings. Since the sequential file 100 is not part of the compressed file sent to the client, Feinman's compressed file does not include any indicator of an

order, much less a predetermined order, in which the application programs are to be installed. The Examiner admits to this point in the Office Action since, as noted above, it is stated in the Final Office Action that “the automatic installations system builds a command for the remote submission...” (Office Action, page 27, emphasis added). This command is not part of the compressed file sent to the client.

Furthermore, in the October 16<sup>th</sup> Amendment, it was submitted that Halpern does not disclose or suggest that the requested plurality of objects are packed into the packed object prior to receiving the request for the plurality of objects. In response, the Examiner now relies on col. 2, lines 34-45 and col. 3, lines 1-25 of Feinman to disclose this feature (Office Action, page 30). Appellants respectfully disagree.

Appellants note that Feinman is directed to a system and method for remotely transferring and installing client server application programs from a source computer onto a remote client within a data processing system in an unattended mode (*see* Feinman: Abstract, col. 1, lines 43-59, and FIG. 1A, e.g., Feinman states that there is a need for an improved method of installation, where the source machine remotely installs the application programs onto another machine and does not require manual interaction at each site). That is, there is no request at all in Feinman.

For instance, in step 12 of FIG. 1a of Feinman (which is the first step of Feinman’s method), the automated installation system packages the one or more applications into a compressed file. This step is carried out absent any request for the compressed file from the destined remote client. Since there is no request prior to the packaging of the applications in Feinman, it cannot disclose that the requested plurality of objects are packed into the packed object **prior to receiving the request for the plurality of objects** as required by claim 31.

Halpern and the AAPA do not cure these deficient teachings of Feinman. Moreover, the teachings of Feinman would lead one skilled in the art away from the modified server query system of Halpern and the AAPA since Feinman's objective is to transmit the compressed file to the client without any query or request at all.

The Examiner response to these arguments by merely stating in the Advisory Action that "Feinman is not relied upon for this teaching but only for the teaching of pre-packaged (no need to request) data". That is, the Examiner admits that there is no request in Feinman and further, there is no need to request in Feinman. Feinman then cannot possibly teach that the plurality of objects requested for by the client are packed into the packed object prior to receiving the request for the plurality of objects. Feinman cannot teach this feature since, as pointed out above and acknowledged by the Examiner, there is no request for the plurality of objects in Feinman.

Therefore, Appellants respectfully submit that Feinman alone, or in combination with Halpern and the AAPA, does not disclose or suggest the noted features of claim 31. Accordingly, Appellants respectfully request withdrawal of the 35 U.S.C. § 103(a) rejection of claim 31.

Claim 32 is patentable *at least* by virtue of its dependency on claim 31.

### ***Conclusion***

For the reasons discussed above, Appellants respectfully request the Board to reverse the final rejections of the pending claims 1-3, 7-10, 13-15, 19-23, 25-29, 31, 32, and 34-36.

SUPPLEMENTAL APPEAL BRIEF  
Application No.: 09/986,248

Attorney Docket No.: A8506

Unless a check is submitted herewith for the fee required under 37 C.F.R. §41.37(a) and 1.17(c), please charge said fee to Deposit Account No. 19-4880. The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account.

Respectfully submitted,

/Quadeer A. Ahmed/

SUGHRUE MION, PLLC  
Telephone: (202) 293-7060  
Facsimile: (202) 293-7860

WASHINGTON DC SUGHRUE/142133

**46159**

CUSTOMER NUMBER

---

Quadeer A. Ahmed  
Registration No. 60,835

Date: March 26, 2009

**CLAIMS APPENDIX**

CLAIMS 1-3, 7-10, 13-15, 19-23, 25-29, 31, 32, and 34-36 ON APPEAL:

1. A method of requesting and processing a plurality of objects from a server, comprising:

opening a session with at least one server;

searching in the at least one server for an information element based upon a search criteria after the opening of the session with the at least one server;

receiving from at least one server search results displayable on a web page comprising a list identifying occurrences of the information element, wherein at least some of said occurrences of the information element identify objects;

generating for at least two identified objects, requests to the at least one server for obtaining the at least two objects;

packing the plurality of requests for the at least two objects into a packed request message and transmitting the packed request message to the at least one server;

receiving a response message from the at least one server, the response message containing the at least two objects packed into the response message;

ending the session with the at least one server after the receiving the response message; and

automatically unpacking the at least two objects contained in the response message for displaying on the web page.

2. The method of claim 1, further comprising decompressing the at least two unpacked objects.

3. The method of claim 2, wherein the decompression of the at least two unpacked objects is performed automatically in response to receiving the response message.

7. The method of claim 1, further comprising displaying on a browser the at least two unpacked objects in a second order provided in the response message, wherein the second order is different from a first order of receipt of the objects.

8. A method of transferring a plurality of objects from a server to a client, comprising:

receiving a request from the client for the plurality of objects, wherein the plurality of objects are occurrences of an information element provided as a search criteria in a data network and wherein the request, displayable on a web page, comprises a plurality of requests for the respective plurality of objects;

retrieving the plurality of requested objects from one or more object stores;  
automatically packing the retrieved plurality of objects into a response message; and  
transmitting the response message to the client, wherein the response message comprises an indicator of a first order in which the packed objects are to be displayed,  
wherein the first order is different from the order in which the plurality of the objects are transmitted to the client.

9. The method of claim 8, further comprising automatically compressing the retrieved plurality of requested objects prior to packing said objects into the response message.

10. The method of claim 8, further comprising automatically compressing the response message prior to transmitting the response message to the client.

13. A client processor, comprising:  
a communications module configured to open a session with at least one server, configured to receive a response message from the at least one server, the response message containing a plurality of packed objects, wherein the plurality of packed objects are occurrences of an information element provided as a search criteria in a data network, configured to send a request message to the at least one server after opening the session with the at least one server, wherein the request, displayable on a web page, comprises a plurality of requests for the plurality of objects, and configured to end the session after receiving response message;  
an unpacking module configured to automatically unpack from the response message the plurality of packed objects; and  
a browser coupled to the unpacking module, configured to display on the web page the plurality of unpacked objects to a user as search results.

14. The client processor of claim 13, further comprising a decompression module configured to decompress the plurality of unpacked objects.

15. The client processor of claim 14, wherein the decompression module is configured to automatically decompress the plurality of unpacked objects in response to receiving the response message.

19. The client processor of claim 13, further comprising displaying on the browser in the web page to the user as the search results, the plurality of unpacked objects in a second order provided in the response message, wherein the second order is different from a first order in which the plurality of the packed objects are unpacked from the response message.

20. A server processor, comprising:

a communication module configured to receive a request message from a client processor for delivery of a plurality of objects, wherein the plurality of objects are occurrences of an information element provided as a search criteria in a data network and wherein the request message, displayable on a web page, comprises plurality of requests for the respective plurality of objects;

a request processor configured to unpack the requests for the plurality of objects from the request message;

an object access module configured to retrieve the plurality of objects requested by the request processor;

an object packing module coupled to the object access module and configured to automatically pack the plurality of objects retrieved by the object access module into a response message; and

an object delivery module coupled to the object packing module and the communication module and configured to output the response message containing the plurality of packed objects to the client processor,

wherein the response message comprises an indicator indicating a first order in which the packed objects are to be presented, the first order different from the order in which the object packing module packs the plurality of the objects.

21. The server processor of claim 20, further comprising a compression module configured to automatically compress the retrieved plurality of requested objects prior to packing the plurality of objects into the response message.

22. The server processor of claim 20, further comprising a compression module configured to automatically compress the response message prior to transmitting the response message to the client.

23. The server processor of claim 20, wherein the packing module is configured to pack the plurality of objects into the response message in a designated order.

25. A computer-readable medium of instructions for requesting and processing a plurality of objects from a server, comprising:

- program operations for opening a session with at least one server;
- program instructions for searching in the at least one server for an information element based upon a search criteria after the opening the session with the at least one server;
- program instructions for receiving from the at least one web server search results, displayable on a web page, comprising a list identifying occurrences of the information element, wherein at least some of said occurrences of the information element identify objects;
- program instructions for generating for at least two identified objects, requests to the at least one web server for obtaining respective object;

program instructions for packing the plurality of generated requests for the at least two objects into a single request message and transmitting the packed single request message to the at least one web server;

program instructions for receiving a single response message from the server, the response message containing the at least two objects packed into the response message;

program instructions for ending the session with the at least one server after the receiving the single response message; and

program instructions for automatically unpacking the at least two objects contained in the response message.

26. The computer-readable medium of instructions of claim 25, further comprising program instructions for decompressing the at least two unpacked objects.

27. The computer-readable medium of instructions of claim 25, further comprising program instructions for outputting the at least two unpacked objects in a second order indicated in the response message, wherein the second order is different from a first order of receipt of the objects.

28. A computer-readable medium of instructions for transferring a plurality of objects from a server to a client, comprising:

program instructions for receiving a request from the client for the plurality of objects, wherein the plurality of objects are displayable on a web page as search results and are occurrences of an information element provided as a search criteria in a data network and

wherein the request message comprises a plurality of requests for the respective plurality of objects;

program instructions for retrieving the plurality of requested objects from one or more object stores;

program instructions for automatically packing the retrieved plurality of objects into a response message; and

program instructions for transmitting the response message to the client,

wherein the response message comprises an indicator indicating a first order in which the packed objects are to be presented,

wherein the first order is different from the order in which the plurality of the objects are received by the client.

29. The computer-readable medium of instructions of claim 28, further comprising program instructions for automatically compressing the retrieved plurality of requested objects prior to packing said objects into the response message.

31. A method of transferring a plurality of objects from a server to a client, comprising:

receiving a request from the client for the plurality of objects, wherein the plurality of objects are displayable on a web page and are occurrences of an information element provided as a search criteria in a data network and wherein the request message comprises a plurality of requests for the respective plurality of objects;

retrieving from an object store a packed object having a plurality of objects corresponding to the requested plurality of objects, wherein the plurality of objects are packed

into the packed object prior to receiving the request for the plurality of objects and wherein the response message comprises an indicator of a predetermined order in which the packed objects are to be presented; and

transmitting the packed object in a response message to the client.

32. The method of claim 31, wherein the retrieved objects are packed into the response message in a designated order.

34. The method of claim 1, wherein the server and a client communicate with each via an HTTP module, wherein the client comprises a browser for receiving the search criteria and for outputting the search results on the web page and wherein the at least two objects are image objects.

35. The method of claim 1, wherein the web page is displayed on a web browser and the at least two objects are images, and wherein a plugin module that operates with the web browser is provided in a client and wherein the plugin module automatically unpacks the at least two image objects contained in the response message, and provides the at least two unpacked image objects to the web browser for display on the web page.

36. A method of requesting and processing a plurality of objects from a server, comprising:

opening a session with at least one server;

receiving from the at least one web server request results comprising a list identifying occurrences of an information element after the opening of the session with the at least one

server, wherein at least some of said occurrences of the information element identify image objects and wherein the request results are displayable on a web page; generating for at least two identified image objects, requests to the at least one web server for obtaining the respective image object; packing the plurality of requests for the at least two objects into a packed request message and transmitting the packed request message to the at least one web server; receiving a response message from the at least one web server, the response message containing the at least two objects packed into the response message; ending the session with the at least one server after the receiving the response message; automatically unpacking the plurality of objects contained in the response message.

**EVIDENCE APPENDIX:**

No evidence is submitted herewith pursuant to 37 C.F.R. §§ 1.130, 1.131, or 1.132 and no other evidence has been entered by the Examiner and relied upon by Appellant in the appeal.

**RELATED PROCEEDINGS APPENDIX**

No decisions have been identified in Section II. Accordingly, no decisions are submitted herewith pursuant to 37 C.F.R. § 41.37(c)(1)(ii).